

4-13-00

JC625 U.S. PTO  
09/546981

04/11/00

IBM Docket No. RAL9-00-0035

**In the United States Patent and Trademark Office  
Patent Application Transmittal**

*Transmitted herewith for filing is the Patent Application of:*

Inventors(s): Anthony Matteo Gallo, Natarajan Vaidhyanathan, Gail Irene Woodland  
For: METHOD FOR BRIDGING CONTROL POINT DATA FRAMES USING A NETWORK PROCESSOR

**Enclosed are**

17 pages of specification, including 16 claims, plus 5 sheets of drawings.

An assignment of the invention to International Business Machines Corporation, Armonk, New York 10504.

A certified copy of a/an application.

X Declaration and Power of Attorney. NOT FULLY EXECUTED  
PTO-1449 & references

X A return post card

Other:

**Filing Fee Calculation (For Other Than Small Entity)**

Basic Fee:						\$690.00
Claims Fees:	Filed	Limit	Extra		Rate per Extra	
Total claims:	16	20	0		\$18.00	\$0.00
Independent claims:	3	3	0		\$78.00	\$0.00
Multiple Dependent Claim Presented					\$260.00	\$0.00
<b>Total</b>						<b>\$690.00</b>

Please charge Deposit Account 09-0464 for the **Total** set forth above. The Commissioner is authorized to charge payment of any additional filing fees required under 37 CFR §1.16 and any patent application processing fees under 37 CFR §1.17 or to credit any overpayment to the identified account. A duplicate copy of this sheet is enclosed.

**Express Mail Certificate**

Express Mail Label No: EK506752130US

Date: April 11, 2000

I hereby certify that I am depositing the papers identified above with the U.S. Postal Service "Express Mail Post Office to Address" service on the above date, addressed to the Commissioner of Patents and Trademarks, Washington, DC 20231

*Karen Orzechowski*

Karen Orzechowski

BY:

*Joselyn G. Cockburn*

Joselyn G. Cockburn

Attorney of Record Reg. No. 27,069

Date: April 11, 2000

IBM Corporation 972/B656

Intellectual Property Law

PO Box 12195

Res. Tri. Park, NC 27709

Telephone: 919-543- 9036 FAX 919-543-3634

# METHOD FOR BRIDGING CONTROL POINT DATA FRAMES USING A NETWORK PROCESSOR

5

**Anthony M. Gallo, Natarajan Vaidhyanathan, Gail I. Woodland**

## **CROSS REFERENCE TO RELATED APPLICATIONS**

10        This application is related by common inventorship and subject  
matter to co-pending application titled "Local MAC Address Learning  
in Layer 2 Frame Forwarding". Until such time as the foregoing  
application is assigned an application number by the U.S. Patent and  
Trademark Office, it may be referenced by the following  
15        applicant:attorney docket numbers: RAL9-00-0032:0421/00059.  
The listed application is assigned to International Business Machines  
Corporation and is entirely incorporated herein by this reference.

## **BACKGROUND OF THE INVENTION**

20        The present invention relates to the processing of data frames  
being transferred through a network, and more particularly to a  
method for using a network processor to perform network bridging  
functions for data frames destined for or originating from a control  
point in the network.

25        In computer networks data is typically communicated between  
network devices in "frames". In addition to user data, each frame  
typically includes fields containing information for routing the frame

through the network. The fields are used by a system of established protocols which may be thought of as being arranged in a “protocol stack” referred to as the Open Systems Interconnect (OSI) model. At the base of the stack is a Layer 1 (L1) which is the protocol relating to the physical transmission media of the network.

Above L1 is a Layer 2 (L2) used in Ethernet and Token Ring local area networks (LANs). The L2 level correlates unique media access control (MAC) addresses for network devices with physical ports.

A network switch may connect multiple virtual LANs (VLANs) and process frames for transfer within a VLAN or between VLANs. The L2 protocol is used for frame transfers within a VLAN. Operations by the switch at the L2 level are typically referred to as “bridging”. For communication between different VLANs, a Layer 3 (L3) protocol is needed. Operations by a network switch at the L3 level are typically referred to as “routing”.

In existing systems, a network switch 100 typically has the configuration shown in Figure 1. A control point (CP) 101 in the switch comprises a general purpose processor and software including a logical router 102 for L3 operations and a logical bridge 104 for L2 operations. The logical router comprises a routing table 103 and the logical bridge comprises a MAC address database 105 correlating MAC addresses with physical ports. Multiple network processors (NPs) 106 (only one is shown) on the switch include a

plurality of ports 107 for receiving and outputting data frames 108. Network processors such as 106 include specialized software and hardware which enable them to perform fast database look-ups for frame forwarding.

5           In existing systems, certain L3 frames propagated on a network are typically processed by the CP. Examples include ARP (Address Resolution Protocol) and OSPF (Open Shortest Path First) frames. In addition, applications running on the CP itself may generate frames destined for other CPs. For example, control protocols for  
10       setting up forwarding tables in the NPs run from CP to CP. Also, CPs may generate and/or receive SNMP (Simple Network Management Protocol) queries.

          When an incoming frame 108 is sent to the CP or the CP generates a frame, the L3 software in the logical router 102 is  
15       executed to perform a look-up in the routing table 103 to determine a logical router interface corresponding to a destination MAC address in the frame. Then, L2 processing by the logical bridge 105 uses the logical router interface and the destination MAC address to look up the MAC address and port corresponding to the L3 destination  
20       address in the MAC address table 105, so that the frame can be forwarded to the appropriate target network device. Also, MAC address "learning" is performed by the logical bridge in the CP, which adds MAC addresses and ports to the MAC database to facilitate frame forwarding.

As noted above, the CP typically utilizes a general purpose processor. The processor typically executes a high-level code, such as "C". Accordingly, the database look-ups entailed in the foregoing are relatively slow and consume a significant proportion of CP processing cycles. Further, memory in the CP must be allocated for the MAC address database used by the logical bridge.

In view of the above, an improved method of handling frames destined for or originating from the CP is needed.

## **SUMMARY OF THE INVENTION**

According to the present invention, the logical bridging function for frames destined for or originating from the CP is offloaded from the CP to a network processor. In a preferred embodiment, frames destined for or originating from the CP are sent to a network processor directly connected to the CP. The network processor performs all the L2 level bridging operations needed by the CP, including MAC address learning for incoming frames, and destination address look-ups and frame forwarding for frames originating from the CP.

Because, as noted above, network processors include special-purpose software and hardware, they are able to perform the database look-ups and learning entailed in L2 processing at a substantially greater rate than the general-purpose CP, providing higher throughput. Further, processing cycles and memory demands

are offloaded from the CP, enabling it to be used for other functions.

### **BRIEF DESCRIPTION OF THE DRAWINGS**

5 Figure 1 shows an example of a prior art network switch configuration;

Figure 2 shows an example of a network switch according to the present invention;

Figure 3 shows a processing flow for a frame destined for the CP;

10 Figure 4 shows a processing flow for a response frame or frame originating from the CP; and

Figure 5 shows examples of computer-usable storage media for storing computer-executable instructions according to the present invention.

### **DETAILED DESCRIPTION OF THE INVENTION**

15 Figure 2 shows a network switch 200 configured according to the present invention. The switch comprises a plurality (not limited to three) of network processors 204, 205 and 207. A CP 206 (the  
20 switch is not limited to having a single CP) is directly connected to NP 205. The CP typically executes control protocols and downloads forwarding tables in the NPs. Where in prior art switch configurations, a logical bridge 213 was in the CP as shown in Figure 1, in the present invention the logical bridge 213 is in the NP 205.

094694-0440  
000000-000000

The logical bridge comprises computer-executable instructions which when executed by hardware in the NP implement L2 data frame processing according to the present invention. The hardware in the NP includes a plurality of forwarding processors which can perform fast database look-ups, and a specialized processor known as a Guided Tree Handler which manages the MAC address database 214.

Network data frames such as frame 202 enter the switch from a physical medium through a plurality of ports 212 in the NPs. Next, in most switches, L2 processing of the frame is performed by a logical bridge such as 210. L2 processing typically comprises a look-up for the destination address (DA) in the frame in the MAC address database 210, to determine whether to filter, flood, or forward the frame. Filtering refers to dropping or discarding a received data frame when processing of the MAC addresses indicates that the originating network segment and the destination segment are the same. Flooding refers to forwarding a data frame to all ports where a DA may reside, when the address has not been learned and is thus unrecognized. Forwarding refers to sending an incoming data frame having a learned MAC address to the corresponding known port or ports.

MAC address learning, as noted above, refers to a known process for facilitating frame forwarding, wherein the source MAC address in a received frame is recorded in a MAC address database

such as 210 and 214, along with its port of origin. This allows future frames destined for the source address (SA) to be forwarded only to the port on which that address is located. Otherwise, an unrecognized address must be flooded.

5           The look-up on the DA in the frame will also determine whether the frame needs L3 processing by the logical router 208. A status bit the MAC address database in the entry will indicate that if the frame is destined for a particular MAC address, it is L3 traffic.

10           If the frame needs L3 processing it is sent to the logical router 208. The logical router comprises computer-executable instructions which when executed by hardware in the NP implement L3 data frame processing according to the present invention. If the frame needs to go to the CP 206, this is indicated by a look-up in the routing tables 209. It is noted that not every L3 frame needs to be  
15 sent to the CP; some are sent directly to other NPs. If a frame is destined for the CP, the logical router 208 sets a status bit so indicating in a frame header 201 appended to the frame.

20           According to the present invention, the frame is then sent to the NP 205 directly connected to the CP 206, via hardware connections known as "switch fabric". When a frame is sent from one NP to another through the switch fabric, the target NP is said to perform "downside" processing of the frame.

          In downside processing according to the present invention, L2 processing is performed by the logical bridge 213 in the NP 205. The

L2 processing includes MAC address learning. To do the learning, a look-up is performed in the MAC address database 214 for the SA in the frame 202. If it is not found, the SA and its port of origin are added to the MAC address database.

5           Subsequently, the frame is sent to the CP 206. Typically, the CP 206 will generate a response frame 203; usually the DA MAC address will be the SA of the incoming frame. The response frame is sent back to the logical bridge 213, which typically will have just learned the SA, and so the response frame can be forwarded via the  
10       switch fabric directly to the target port (usually in the NP where the original frame 202 entered). If the CP originates the frame 203 or the SA has aged out of the MAC database 214, the frame is flooded.

          Figure 3 illustrates a processing flow for an incoming frame according to the present invention in flowchart form. When a frame  
15       is received, a DA look-up is performed by the logical bridge in the receiving NP, as shown in blocks 300 and 301. If the look-up determines that the frame is an L3 frame, the frame is sent to the logical router as shown in blocks 302 and 304. Otherwise, the frame is forwarded or flooded to the target NP and port, as shown in block  
20       303.

          The logical router does L3 processing, performing a look-up in the routing tables to determine whether the frame needs to go to the CP, as shown in blocks 304 and 305. If not, the frame is sent to the target NP/port (block 306).

If the frame needs to go to the CP, a status bit indicating this is set in a frame header appended to the frame, and the frame is sent for downside processing to the NP directly connected to the CP as shown in block 307.

5       The logical bridge in the NP performs L2 processing of the frame, including MAC address learning. As shown in blocks 308-311, to do the learning, the SA in the frame is looked up in the local MAC address table. If it is already present, the frame is sent to its destination, the CP. If not, the SA and its port of origin is added to  
10      the MAC address database, and the frame is sent to the CP.

Figure 4 shows a flow for a response frame or frame originating from the CP. The frame is sent from the CP to the logical bridge, which performs a look up in the MAC address table for the DA, as shown in blocks 400 and 401. Typically, the DA will have been  
15      recently learned, and the frame will be forward to the target NP/port, as shown in blocks 402 and 403. Otherwise, the frame will be flooded or discarded (block 404).

It may be appreciated from the foregoing that the present invention frees up a significant proportion of CP processor cycles and  
20      memory that would otherwise have been used for logical bridging functions, allowing the CP to be used for other work. Additionally, the database look-ups necessitated by bridging are performed significantly faster, by the specialized hardware and software of the NPs.

As noted above, in a preferred embodiment the invention is implemented as a configuration of specialized software comprising computer-executable instructions in a network switch. As shown in Figure 5, software executed by the NPs is typically known as "picocode" 505, executable by specialized processors as detailed above. The CP 206 is typically a general purpose processor that executes a high-level code 504 such as "C" language code. The picocode 505 and CP code 504 may be stored and transported on computer-usable media such as diskette 500, CD-ROM 501, magnetic tape 502 or fixed disk 503. In a well-known fashion, the code may be retrieved from the computer-usable media, loaded into a memory on the switch and executed to effect the features of the invention.

The foregoing description of the invention illustrates and describes the present invention. Additionally, the disclosure shows and describes only the preferred embodiments of the invention, but it is to be understood that the invention is capable of use in various other combinations, modifications, and environments and is capable of changes or modifications within the scope of the inventive concept as expressed herein, commensurate with the above teachings, and/or the skill or knowledge of the relevant art. The embodiments described hereinabove are further intended to explain best modes known of practicing the invention and to enable others skilled in the art to utilize the invention in such, or other, embodiments and with the various



### **What is Claimed Is:**

1                   1.     In a network switch comprising a control point and a  
2     plurality of network processors, a method comprising:

3                   (a) receiving data frames from a network; and  
4                   (b) performing logical bridging of data frames destined  
5     for or originating from said control point in a network processor  
6     directly connected to said control point.

1                   2.     The method of claim 1, said step (b) comprising:

2                   (c) determining whether said data frame is destined for  
3     said control point; and

4                   (d) sending said data frame to said network processor  
5     directly connected to said control point when said step (c) indicates  
6     that said data frame is destined for said control point.

1                   3.     The method of claim 2, said step (c) comprising:

2                   (e) looking up a destination address in said frame in a  
3     media access control (MAC) address database;

4                   (f) sending said data frame to a logical router when said  
5     look-up determines that said data frame requires processing by a  
6     logical router;

7                   (g) looking up a destination address in a routing table in  
8     said logical router; and

9                   (h) sending said frame to said network processor directly

10 connected to said control point when said look-up determines that  
11 said frame is destined for said control point.

1 4. The method of claim 3, further comprising:  
2 setting a bit in a frame header appended to said frame to indicate  
3 that said frame is destined for said control point.

1 5. The method of claim 1, said step (b) comprising:  
2 learning a source MAC address in said frame in a MAC  
3 address database; and  
4 sending said frame to said control point.

1 6. The method of claim 1, said step (b) comprising:  
2 looking up a destination address in a frame originating  
3 from said control point in a MAC address database; and  
4 forwarding said frame to a target network processor and  
5 port found in said look-up.

1 7. A network switch comprising:  
2 a control point;  
3 a plurality of network processors;  
4 said plurality of network processors programmed with  
5 logical bridging and logical routing functions;  
6 wherein a network processor directly connected to said  
7 control point performs logical bridging functions needed by said

8 control point.

1 8. The network switch of claim 7, wherein said logical  
2 bridging and logical routing functions determine that an incoming  
3 data frame to one of said plurality of networks processors is destined  
4 for said control point and send said data frame to said network  
5 processor directly connected to said control point.

1 9. The network switch of claim 8, wherein said logical  
2 bridging function in said network processor directly connected to said  
3 control point learns a source address in said frame in a MAC address  
4 database.

1 10. The network switch of claim 9, wherein said logical  
2 bridging function in said network processor directly connected to said  
3 control point receives a frame originating from said control point,  
4 looks up said learned source address, and forwards said frame  
5 originating from said control point to a target network processor  
6 corresponding to said learned source address.

1 11. A computer-usable medium storing computer-  
2 executable instructions, said instructions when executed by  
3 processors in a network switch comprising a control point and a  
4 plurality of network processors, implementing a method comprising:  
5 (a) receiving data frames from a network; and

6 (b) performing logical bridging of data frames destined  
7 for or originating from said control point in a network processor  
8 directly connected to said control point.

1 12. The computer-usable medium of claim 11, said step (b)  
2 comprising:

3 (c) determining whether said data frame is destined for  
4 said control point; and

5 (d) sending said data frame to said network processor  
6 directly connected to said control point when said step (c) indicates  
7 that said data frame is destined for said control point.

1 13. The computer-usable medium of claim 12, said step  
2 (c) comprising:

3 (e) looking up a destination address in said frame in a  
4 media access control (MAC) address database;

5 (f) sending said data frame to a logical router when said  
6 look-up determines that said data frame requires processing by a  
7 logical router;

8 (g) looking up a destination address in a routing table in  
9 said logical router; and

10 (h) sending said frame to said network processor directly  
11 connected to said control point when said look-up determines that  
12 said frame is destined for said control point.

1                   14. The computer-usable medium of claim 13, said  
2 method further comprising:  
3                   setting a bit in a frame header appended to said frame to  
4 indicate that said frame is destined for said control point.

1                   15. The computer-usable medium of claim 11, said step  
2 (b) comprising:  
3                   learning a source MAC address in said frame in a MAC  
4 address database; and  
5                   sending said frame to said control point.

1                   16. The computer-usable medium of claim 11, said step (b)  
2 comprising:  
3                   looking up a destination address in a frame originating  
4 from said control point in a MAC address database; and  
5                   forwarding said frame to a target network processor and  
6 port found in said look-up.

# METHOD FOR BRIDGING CONTROL POINT DATA FRAMES USING A NETWORK PROCESSOR

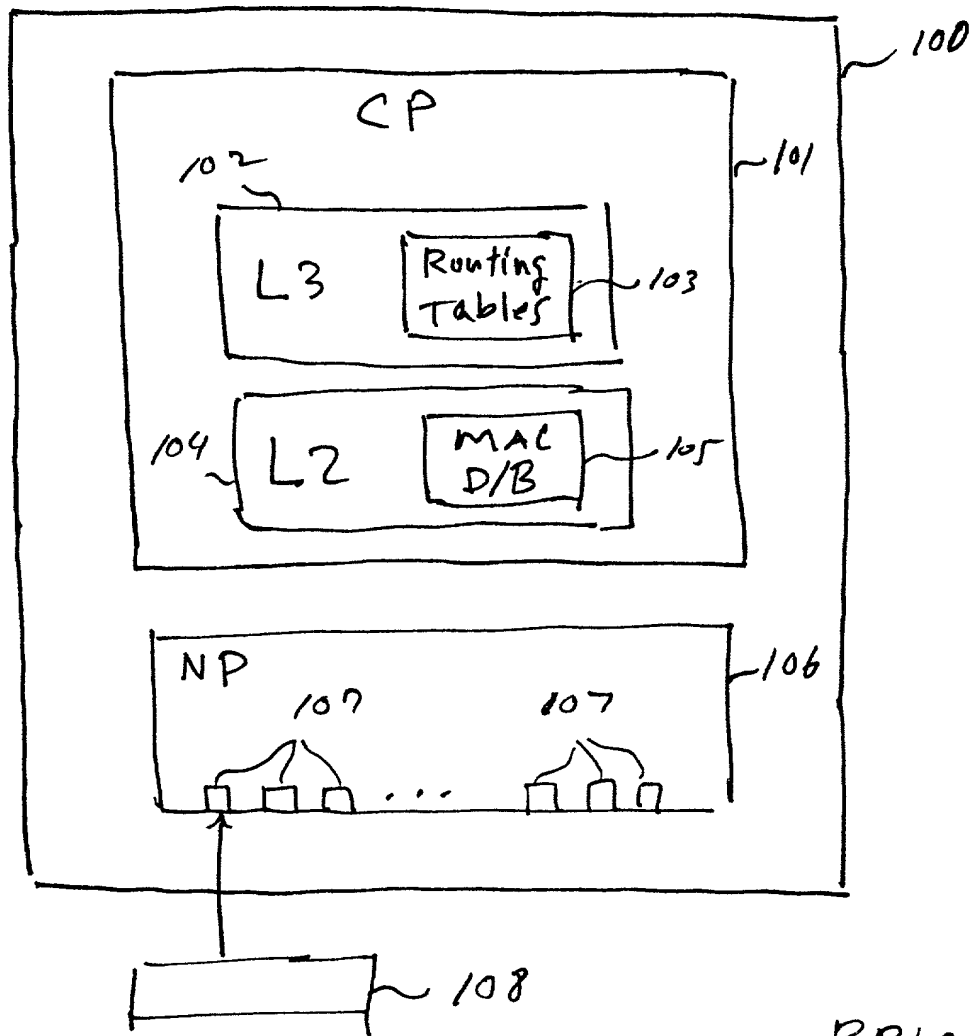
## ABSTRACT OF THE INVENTION

5

Certain Layer 3 protocol data frames propagated on a network are typically processed by a control point (CP) in a network switch. The logical bridging and routing functions required in this processing typically entail network device address look-ups in routing tables and address databases. Using the CP to perform these look-ups is expensive in terms of processor cycles and memory. To offload the CP, the bridging functions are performed a network processor in the switch. The network processor has specialized software and hardware enabling it to perform the required database look-ups faster and more efficiently than the CP.

10

15



PRIOR ART

FIG. 1



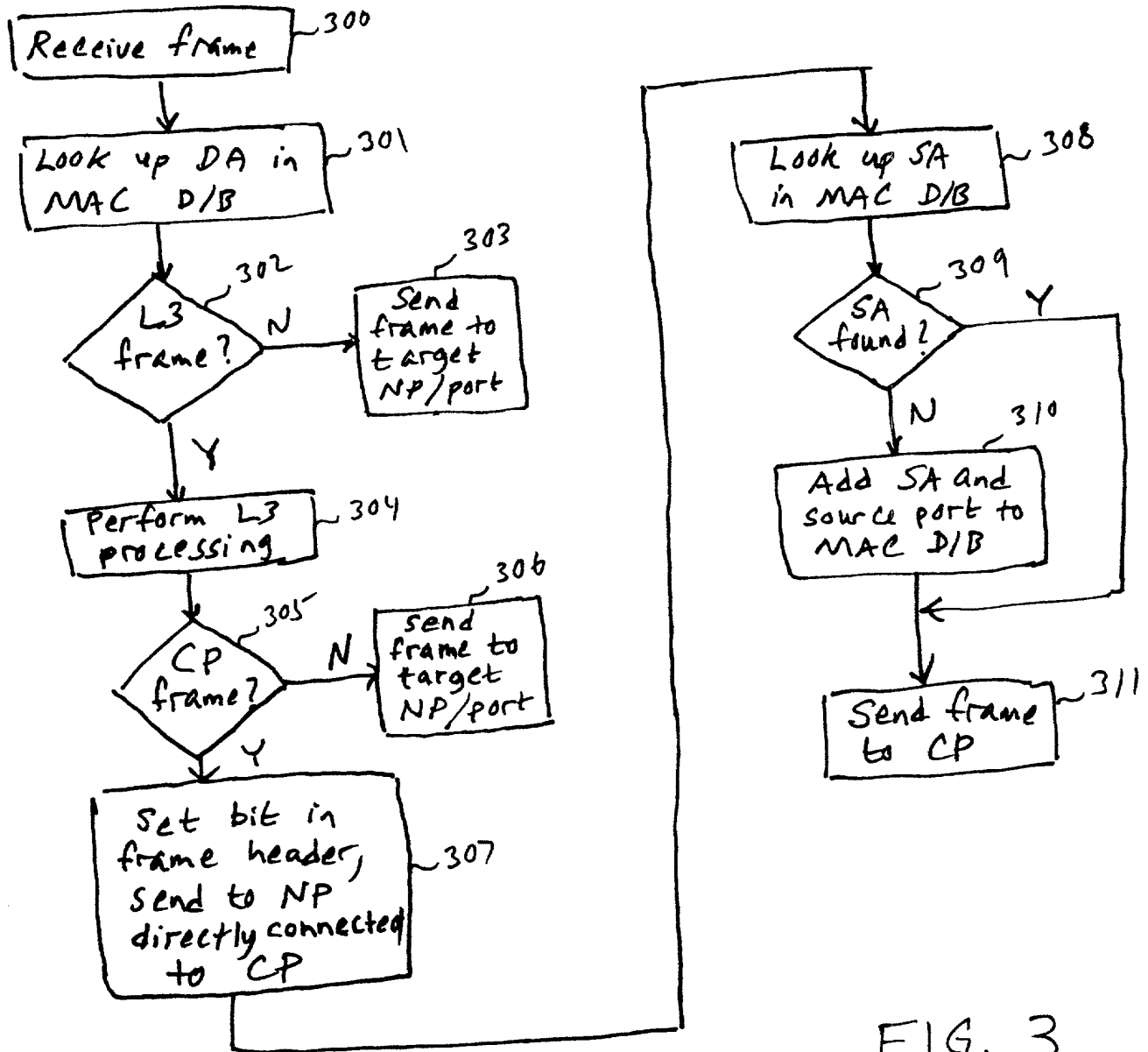


FIG. 3

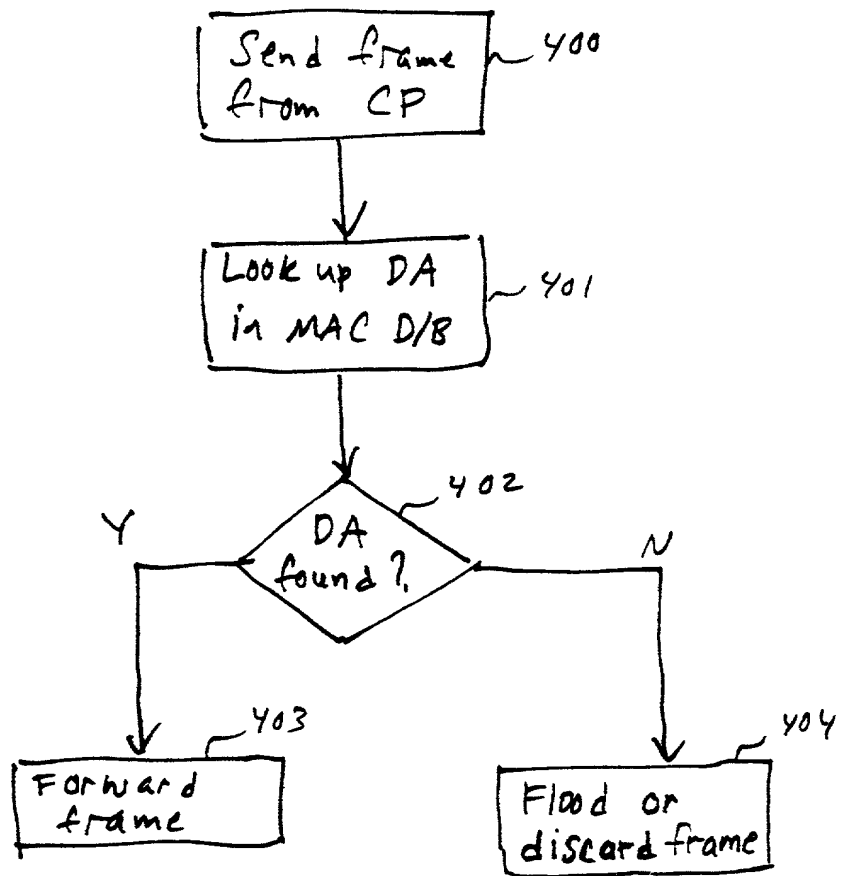


FIG. 4

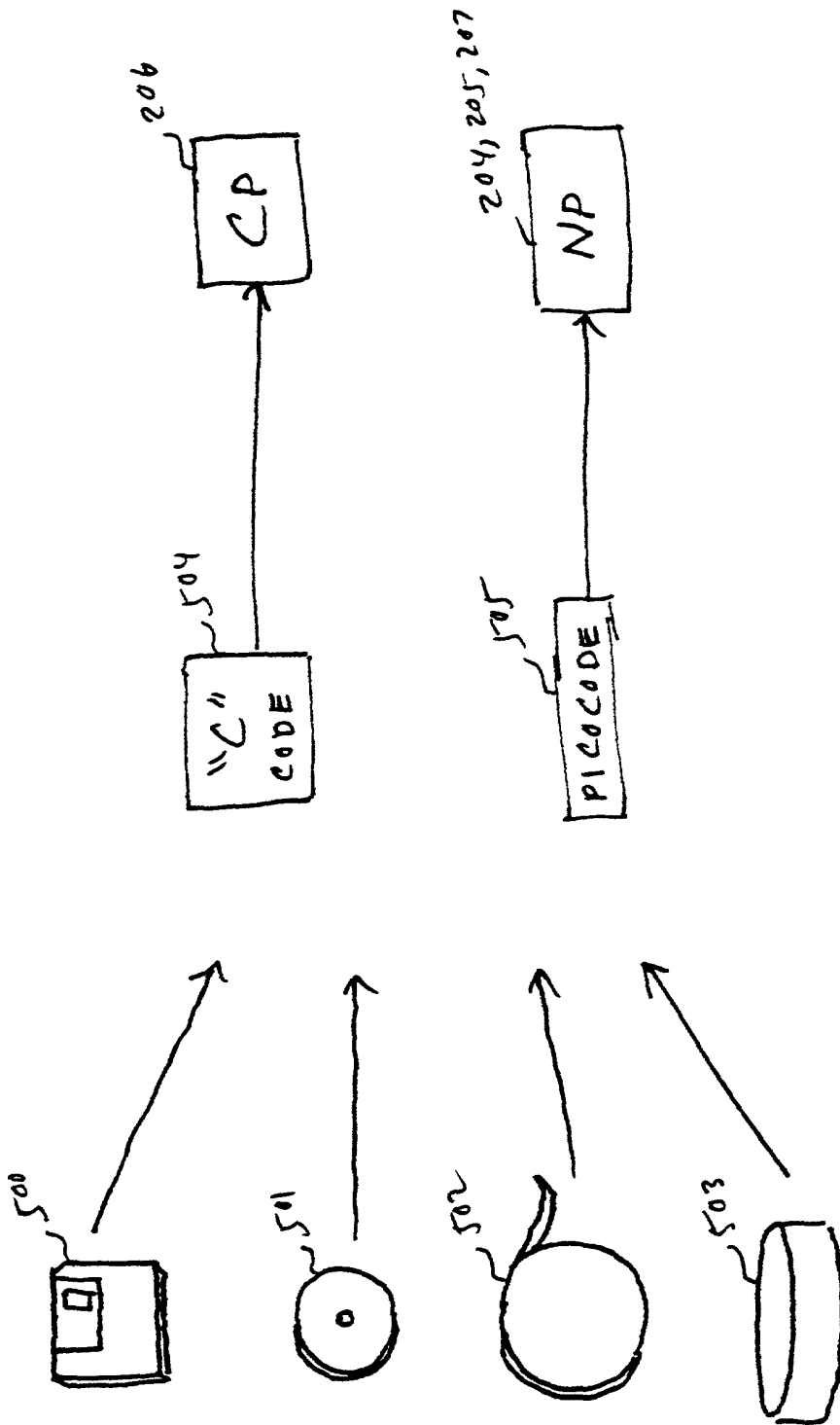


FIG. 5

**DECLARATION AND POWER OF ATTORNEY  
FOR PATENT APPLICATION**

As a below named inventor, I hereby declare that:

My residence, post office address and citizenship are as stated below next to my name; I believe I am an original, first and joint inventor of the subject matter which is claimed and for which a patent is sought on the invention entitled:

**METHOD FOR BRIDGING CONTROL POINT DATA FRAMES USING A  
NETWORK PROCESSOR**

the specification of which is identified by the attorney (IBM) Docket Number appearing above.

I hereby state that I have reviewed and understand the contents of the above- identified specification, including the claims.

I acknowledge the duty to disclose information which is material to the patentability of this application in accordance with Title 37, Code of Federal Regulations, §1.56.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

**Prior Foreign Application(s)**

<u>Number</u>	<u>Country</u>	<u>Day/Month/Year</u>	<u>Priority Claimed</u>
---------------	----------------	-----------------------	-------------------------

I hereby claim the benefit (a) under Title 35, United States Code, §119(e) of any U.S. application listed below and identified as a provisional application or (b) under Title 35, United States Code, §120 of any U.S. application listed below and not identified as a provisional application, and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior U.S. application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose information material to the patentability of this application as defined in Title 37, Code of Federal Regulations, §1.56 which occurred between the filing date of the prior application and the national or PCT international filing date of this application

**Prior U.S. Applications**

<u>Serial No.</u>	<u>Filing Date</u>	<u>Status</u>
-------------------	--------------------	---------------

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

As a named inventor, I hereby appoint the following attorneys and/or agents to prosecute this application and transact all business in the Patent and Trademark Office connected therewith: Daniel E. McConnell, Reg. No. 20,360; Kenneth A. Seaman, Reg. No. 28,113; Joscelyn G. Cockburn, Reg. No. 27,069; Gerald R. Woods, Reg. No. 24,144; John D. Flynn, Reg. No. 35,137; Horace St. Julian, Reg. No. 30,329; Joseph C. Redmond, Jr., Reg. No. 18,753; John E. Hoel, Reg. No. 26,279; Christopher A. Hughes, Reg. No. 26,914; and Edward A. Pennington, Reg. No. 32,588.

Send all correspondence to: Joscelyn G. Cockburn, IBM Corporation 972/B656; PO Box 12195; Research Triangle Park, NC 27709.

**First Inventor:** Anthony Matteo Gallo

**Signature:** \_\_\_\_\_ **Date** \_\_\_\_\_

**Residence:** 3308 Corsham Drive, Apex, NC 27502

**Citizenship:** USA

**Post Office Address:** same as above

**Second Inventor:** Natarajan Vaidhyanathan

**Signature:** \_\_\_\_\_ **Date** \_\_\_\_\_

**Residence:** 3611-13K University Drive, Durham, NC 27707

**Citizenship:** India

**Post Office Address:** same as above

**Third Inventor:** Gail Irene Woodland

**Signature:** \_\_\_\_\_ **Date** \_\_\_\_\_

**Residence:** 123 Booth Road, Chapel Hill, NC 27516

**Citizenship:** USA

**Post Office Address:** same as above